

Serial No. 10/645,567

JUN 29 2006

AMENDMENTS TO THE CLAIMS

2

Please cancel claims 1-2.

4. Please amend claims 3, 4, and 11 as follows:

Claims 1-2 (cancelled).

Claim 3 (currently amended): ~~The apparatus of claim 1, further comprising:~~

2

An apparatus for verifying correctness of a behavioral model of a microcode machine, the microcode machine operable in a native state and an emulated state, the apparatus, comprising:

4

means for producing the native state;

6

means for producing the emulated state;

means for comparing the native state and the emulated state;

8

means for determining a microcode entry point into a microcode storage device that stores microcode; and

10

means for generating microinstructions corresponding to the microcode, wherein a sequence of microinstructions corresponds to a macroinstruction, wherein the means for producing the native state includes means for executing the microinstructions, [[and]] wherein the means for producing the emulated state includes means for executing the macroinstruction; and wherein the behavioral model comprises one of a hardware description language and a processor.

14

Serial No. 10/645,567

Claim 4 (currently amended): A method for verifying the correctness of a
processor behavioral model, wherein a processor operates in one of a native mode
state and an emulated mode state, the method, comprising:

determining if a macroinstruction to be executed is a native instruction;
if the macroinstruction is a native instruction, executing the native
instruction, the execution producing the native mode state of the processor;
if the macroinstruction is not a native instruction:
fetching the macroinstruction,
providing microinstructions corresponding to the macroinstruction,
and
executing the microinstructions, the execution producing the native
mode state of the processor;
executing the macroinstruction, the execution producing an emulated state
of the processor; and
comparing the native mode state [[the]] of the processor with the emulated
state of the processor.

Claim 5 (original): The method of claim 4, further comprising checking the native
mode state of the processor against the behavioral model.

Claim 6 (original): The method of claim 5, wherein checking the native mode
state of the processor against the behavioral model comprises checking for faults
generated during execution of the native instruction.

Serial No. 10/645,567

Claim 7 (original): The method of claim 4, wherein providing microinstructions
2 comprises:

generating a microcode entry point, the microcode entry point indicating a
4 location in a microcode storage where the microinstructions begin; and
expanding the microcode entry point.

Claim 8 (original): The method of claim 7, wherein expanding the microcode
2 entry point comprises:

reading the microcode storage;
4 generating alias fields for the microinstructions;
generating a control word; and
6 generating the microinstructions.

Claim 9 (original): The method of claim 8, wherein generating the
2 microinstructions comprising determining one or more of faults, traps, and
exceptions.

Claim 10 (original): The method of claim 4, further comprising:

2 testing the fetched macroinstruction for faults; and
if a fault is detected, generating a fault entry point into a microcode storage
4 that stores microcode for use by the processor.

Claim 11 (currently amended): The method of claim 4, wherein executing the
2 macroinstruction comprises executing the macroinstruction on an emulated mode

Serial No. 10/645,567

reference simulator that verifies the microinstructions correctly emulate[[s]] the
macroinstruction.

Claim 12 (original): A method for verifying the correctness of a behavioral model
of a micro-coded machine, comprising:

executing a sequence of microinstructions on a native mode simulator, the
execution producing a native mode state of the micro-coded machine, wherein the
sequence of microinstructions corresponds to a macroinstruction;

executing the macroinstruction on an emulated mode reference simulator,
the execution producing an emulated state of the micro-coded machine; and

checking the native mode state and the emulated state against the
behavioral model.

Claim 13 (original): The method of claim 12, further comprising:

determining the sequence of microinstructions, comprising:

reading the macroinstruction, and

determining an entry point into a microcode storage, the microcode
storage storing microinstructions for execution by the micro-coded machine,
wherein the entry point defines a first microinstruction in the microinstruction
sequence; and

expanding the microcode entry point.

Claim 14 (original): The method of claim 13, wherein expanding the microcode
entry point comprises:

reading the microcode storage;

Serial No. 10/645,567

4 generating alias fields for microinstructions in the sequence of
microinstructions;

6 generating a control word; and
 generating the microinstructions.

Claim 15 (original): The method of claim 14, wherein generating the
2 microinstructions comprises determining one or more of faults, traps, and
exceptions.

Claim 16 (original): The method of claim 12, further comprising:
2 testing the macroinstruction for faults; and
 if a fault is detected, generating a fault entry point into the microcode
4 storage.

Claim 17 (original): An apparatus that verifies the correctness of a processor
2 behavioral model, comprising;
 a microcode storage that stores microcode corresponding to
4 microinstructions;
 a microcode expander that reads the microcode storage;
6 a native mode reference simulator that executes microinstructions, wherein
execution of the microinstructions produces a native mode state of a processor;
8 an emulated mode reference simulator that executes macroinstructions,
wherein a macroinstruction comprises a sequence of microinstructions, and
10 wherein execution of the macroinstruction produces an emulated mode state of the
processor; and

Serial No. 10/645,567

12 a state checker that compares the native mode state and the emulated mode
state to the behavior model.

2 Claim 18 (original): The apparatus of claim 17, further comprising an emulated
instruction sequencer, wherein the native mode reference simulator reports faults
and state or control changes to the emulated instruction sequencer, and wherein
4 the emulated instruction sequencer determines a subsequent microinstruction for
execution based on one or more of the faults and the state or control changes.

2 Claim 19 (original): The apparatus of claim 17, wherein the processor behavioral
model is written in hardware description language.

2 Claim 20 (original): The apparatus of claim 17, wherein the emulated mode
reference simulator is independent of the processor behavioral model.